

Python Questions and Answers- 75

အမှာစာ

ယခုစာအုပ်သည် ကျနော်ရဲ့ Python for beginners စာအုပ်အပြီး တော်တော်ကြာမှပြန် ထုတ်ဖြစ်တဲ့ စာအုပ်လေးပါ။ ဒီစာအုပ် က Questions တွေလဲမေးထားသလို ပြန်လည်ဖြေပြထားပါတယ်။ဒီစာအုပ်မှာ Python ရဲ့ module တော်တော်များများကို အသုံးပြုထားပါတယ်။ ထို့ကြောင့်လုပ်ငန်းခွင်ဝင်နေသူများနဲ့ တခြား ဆက်မိစပ်ရာ လေ့လာသူတွေအတွက်လည်း စာအုပ်ကောင်း တစ်အုပ်ဖြစ်ပါလိမ့်မယ်။

Python for beginners → [Download Link](#)

လေးစားစွာဖြင့်

A-Coder

Simple Projects

ဒီအပိုင်းမှာ ကျနော်အနေနဲ့ ရိုးရိုး Projects အသေးလေးတွေ တစ်နည်းအားဖြင့် လုပ်ငန်းခွင်မှာ အသုံးဝင်မဲ့ parts လေးတွေကို ရေးပြမှာဖြစ်ပါတယ်။ ဒီအပိုင်းပြီးရင် တွေးတက်ခေါ်တက်လာမှာပါ။

Q-1

Bank account list တစ်ခုရှိတယ်။ Bank account တွေက မတူတဲ့ နိုင်ငံတွေကနေဖွင့်ထားတာ။ Bank account ရဲ့ နောက်ဆုံးမှာ currency name လေးတွေပါတယ်။ User က USD လို့ရိုက်ထည့်လိုက်ရင် USD နဲ့ ပတ်သတ်တဲ့ bank account ကို ပြပေးရမယ်။

Q-2

စာတန်းရှည်ပါတဲ့ variable တစ်ခုဖန်တီးပါ။ ထို့နောက် output ထုတ်သည့်အခါ ထို့ စာတန်းထဲရှိ white space များအားဖျောက်ပေးပါ။

Q-3

Random Program တစ်ခုရေးရမည်ဖြစ်သည်။ User ထည့်လိုက်တဲ့ data နဲ့ random number နဲ့ မညီမခြင်း user ကို Input သွင်းခိုင်းရမည်ဖြစ်သည်။

Q-4

HTML ကို python web server ဖြင့် run ပါ။(ဆိုလိုတာက html page ကို python နဲ့ browser မှာပေါ်အောင် run ခိုင်းတာပါ။)

Q-5

Hide the credit card number. Program က ရှင်းပါတယ်။ User ထည့်လိုက်တဲ့ card အစ 4 လုံးနောက် က ဟာတွေကို * character အနေနဲ့ ပြပေးရပါမယ်။

Q-6

Discount Price. ဒီ Program ကတော့ customer က ကိုယ်ဆီကနေ ၁၀သိန်းအထက် ဝယ်ရင် cash back or discount amount နဲ့ပြပေးရမယ်။ ၁သောင်း ပြန်ပေးမယ်ဆို original amount ရော၊ discount amount နဲ့ total amount သုံးခုလုံး ကို ပြပေးရမှာဖြစ်ပါတယ်။

Q- 7

ဒီတစ်ခါလဲ calculation လေးပဲ။ ဘာလုပ်ခိုင်းမှာလဲ ဆိုတော့ ရောင်းခဲ့တဲ့ ဈေး amount တွေကို list တစ်ခုအနေနဲ့ တည်ဆောက်ထားပါ။အဲ့ထဲကနေ အများဆုံး amount ကို output ထုတ်ပေးပါ။

Q-8

Integer တွေစီပဲထည့်ထားတဲ့ List တစ်ခုတည်ဆောက်ပါ။ ထို List ထဲရှိ Integer တွေကို string ပြောင်းပြီး string list တစ်ခုအဖြစ် output ထုတ်ပေးပါ။

Q-9

Question -9 ကို ပဲ Function နဲ့ရေးပြရမှာပါ။ (9 မှာထဲက function နဲ့ရေးထားရင် ပိုတော်တာပေါ့။)

Solution
S-1

```
bank_acc=['123423454(USD)', '12345234343(MMK)', '123443456543(EUR) ']  
currency_type=input("Enter Currency")  
for bank in bank_acc:  
    if bank[-4:-1]==currency_type:  
        print(bank[:-5])  
✓ 5.0s  
123423454
```

Descriptions

Bank Acc List တစ်ခုကိုတည်ဆောက်ထားလိုက်ပါ။ ထို့နောက် Currency Type input တစ်ခုကိုရေးသားထားပါတယ်။ For loop နဲ့ bank acc တွေကို Loop ပတ်ပြီး Bank acc ထဲ က currency type နဲ့ user ထည့်လိုက်သော currency type နဲ့ ညီတဲ့ bank acc ကို ပဲ output ထုတ်ပေးမှာဖြစ်ပါတယ်။ Output ထုတ်တဲ့ အခါ မှာ currency type ပါမလာတာကိုး သတိထားမိမှာပါ။ ဒီအကြောင်းကို နားလည်ချင်ရင် ကျနော်ရဲ့ first edition python basic စာအုပ်ကိုရှာဖတ်ဖို့ အကြံပေးပါရစေ။

S-2

```
b="Written By A-Coder"  
print(b.replace(" ", ""))  
✓ 0.0s  
WrittenByA-Coder
```

Descriptions

အဓိက ကတော့ replace function ကို အသုံးပြုထားတာပဲဖြစ်ပါတယ်။ ပထမဆုံး " " က string က space s တွေကို ဖမ်းပ

S-3

```
import random  
n = random.randrange(1,5)  
guess = int(input("Enter any number: "))  
while n!= guess:  
    if guess < n:  
        print("Too low")  
        guess = int(input("Enter number again: "))  
    elif guess > n:  
        print("Too high!")  
        guess = int(input("Enter number again: "))  
    else:  
        break  
print("you guessed it right!!")  
[2] ✓ 2.2s  
... you guessed it right!!
```

Descriptions

ဒီအပိုင်းမှာတော့ Python ရဲ့ random module ကို အသုံးပြုထားပါတယ်။ Random ဆိုတဲ့ program က ကျနော်တို့ သတ်မှတ်ထားတဲ့ range အတွင်း သူကျ ချင်တဲ့ Number ကျပေးမှာပါ။ အဲ့ကျပေးတဲ့ Number နဲ့ ကျနော်တို့ ထည့်လိုက်တဲ့ Number နဲ့ အတူမခြင်း program က run နေမှာပါ။

S-4

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Document</title>
7 </head>
8 <body>
9   <h2>Welcome Python</h2>
10 </body>
11 </html>
```

OUTPUT DEBUG CONSOLE PROBLEMS TERMINAL sudo + v [] [] ... ^

```
root@dev:/home/dev/project/student_management/templates# python3 -m http.server 9000
Serving HTTP on 0.0.0.0 port 9000 (http://0.0.0.0:9000/) ...
```



Welcome Python

Descriptions

Flask နဲ့ run မယ်လို့တွေးမိကြသေးလားဗျ။ Python ရဲ့ Http server ကို အသုံးပြုပြီး html page အောက်မှာ run ပြထားတာဖြစ်ပါတယ်။

S-5

```
bank_acc='123343534545444'
print(bank_acc[:3],"*****")
```

✓ 0.0s

123 *****

Description

Bank Acc number ကို Hidden လုပ်တဲ့ program လေးဖြစ်ပါတယ်။ ရှိရင်းပါတယ်။ Element Index Number 3 မှာ break လိုက်ပြီး နောက် ကဟာတွေ ကို * နဲ့ပေါင်းလိုက်တာပါ။ ကျနော်တို့အရမ်းတွေ့ခဲ့ရတဲ့ Program လေးတစ်ခုလဲဖြစ်ပါတယ်။

S-6

```
customer_price=int(input("Total Pay "))
discount=10000
if customer_price>=100000:
    total=customer_price-discount
    print(f"""
        Total Amount {customer_price}
        discount Amount {discount}
        Pay amount{total}
    """)
else:
    print(f'Total Pay {customer_price}')
```

[9] ✓ 4.4s

...

Total Amount 1000000
discount Amount 10000
Pay amount990000

Descriptions

Program ကိုတွေ့အထူးတလည်ရှင်းပြစရာလိုမယ်မထင်ပါဘူး။ User ထည့်လိုက်တဲ့ amount ကို condition စစ်တယ်။ ကိုက်ညီတဲ့ ကောင်ကိုး အလုပ်လုပ်သွားတယ်။ f ‘ ကိုတွေ့တချို့တွေသိမယ်မထင်ဘူး ဒါကြောင့် ထပ်လေ့လာလိုက်ပါ။

S-7

```
price = [500,100,13000,400]
max_amount = max(price)
print(max_amount)
```

Descriptions

Price List တစ်ခုလုပ်ထားတယ်။ အထဲမှာ amount တွေထည့်ထားတယ်။ line2 က max function က အဲ့ amount တွေထဲက အများဆုံးကောင်ကိုစစ်ထုတ်ပေးပဲ။

S-8

```
a = [4,5,5,7]
print([str(x) for x in a])
```

✓ 0.0s

['4', '5', '5', '7']

Descriptions

ဟုတ်ပါတယ် နှစ်ကြောင်းတည်းသုံးသွားတာပါ။ဒီနေရာကျနော်လဲအစက for loop ကြီးနဲ့ပတ်ပြတာခြင်း အတူတူ Python ရဲ့ One liner for loop နဲ့ ရေးတဲ့ နည်းကို သတိရသွားလို့။ တချို့ beginner တွေတွေသိမယ်မထင်ဘူး။ ဒါကြောင့်ထည့်ရေးဖို့စဉ်းစားလိုက်တာဖြစ်ပါတယ်။ သူက output ကို list အနေနဲ့ ပဲ ပြန်ထွက်ပါတယ်။ အောက် က program နဲ့ တူတူပါပဲ။အရှေ့မှာ str(x) ထည့်လိုက်သောကြောင့် item တွေက string datatype တွေ ဖြစ်ကုန်ဖြစ်ပါတယ်။

```
x=[]
for i in range(0,5):
    x.append(i)
print(x)
```

✓ 0.0s

[0, 1, 2, 3, 4]

S-9

```
def convert(x):  
    return [str(i) for i in x]  
convert([1,2,3])  
  
✓ 0.0s  
['1', '2', '3']
```

Solution 9 က 8 ကို သေချာနားလည်ရင်ရေးလိုရပါတယ်။ Function တစ်ခုတည်ဆောက်ပြီး argument တစ်ခုထည့် လိုက်တယ်။ ပြီးတော့ return ထားတယ်။ အဲ့အောက်မှာ List parameter တစ်ခုထည့်ထားလိုက်တယ်။ ဒီလောက်ပါပဲ။

Part-2

Q-10

Check your os system . သင်အသုံးပြုနေတဲ့ လက်ရှိ os အမျိုးအစားကို output ထုတ်ပြပါ။

Q-11

Random Password Generator တစ်ခု ပြုလုပ်ဖန်တီးပါ။

Q-12

User ထည့်လိုက်တဲ့ Input Password ကို MD5 အဖြစ် ပြောင်းပြီး ထို value ကို output ထုတ်ပေးပါ။

Q-13

လက်ရှိရောက်နေသော ခုနှစ်ပေါ်မူတည်ပြီး user ရဲ့ အသက်ကို တွက်ပြပါ။ (today – user's birthday)

Q-14

Image ကို white and black ပြောင်းပါ။

Q-15

မိမိစက်ရဲ့ လက်ရှိ IP Address ကို output ထုတ်ပြပါ။

Q-16

Screen ပေါ် cursor ထောက်လိုက်တာနဲ့ Script By A-Coder ကို auto ရိုက်ပေးပါ။

Q-17

Phone Number ထည့်လိုက်ရုံနဲ့ ဘယ်နိုင်ငံလဲ ဆိုတာ output ထုတ်ပေးပါ။

Q-18

User Information Dictionary Data Type တစ်ခု ဖန်တီးပါ။ထို့နောက် User က ရှာလိုက်တဲ့ လူရဲ့ အချက်အလက်ကို ထုတ်ပြပေးပါ။

Q -19

Website တစ်ခုရဲ့ title tag မှာ ရေးထားတဲ့ စာသား ကို output ထုတ်ပြပါ။

Solutions

S-10

```
import platform
platform.system()

[5] ✓ 0.0s

... 'Linux'
```

Description

System ရဲ့ Details ကို စစ်ကြည့်ဖို့အတွက် Python မှာ Platform module ရှိပါတယ်။ ယခုက ထို platform module ကို အသုံးပြုပြီး system ရဲ့ OS အမျိုးအစားကို ထုတ်ကြည့်ချင်းဖြစ်ပါတယ်။

S-11

```
import string
import secrets
letters = string.ascii_letters
digits = string.digits
special_chars = string.punctuation
alphabet = letters + digits + special_chars
pwd_length = int(input("Enter Total Password Number: "))
pwd = ''
for i in range(pwd_length):
    pwd += ''.join(secrets.choice(alphabet))

print("Secure Password is",pwd)

[3] ✓ 2.4s

... Secure Password is rV\0N7$9L/o9
```

Description

String နဲ့ secrets module နှစ်ခုကို အသုံးပြုထားပါတယ်။ အောက် ၄ ကြောင်းကတော့ letters and digits စတာတွေကိုး သတ်မှတ်တာဖြစ်ပါတယ်။ Alphabet ကတော့ စကားလုံးတွေအကုန်ပေါင်းထားတာဖြစ်ပါတယ်။ နောက်တစ်ကြောင်း က user ထားချင်တဲ့ passwords အရေအတွက်ကို ထည့်ခိုင်းပြီး အောက်ဆုံး code တွေကျတော့ random ရွေးပြီး output ထုတ်ပေးတာဖြစ်ပါတယ်။

S-12

```
import hashlib
user_password=input("Enter Password")
print(hashlib.md5(user_password.encode('utf-8')).hexdigest())
```

[7] ✓ 3.2s

... 6b76b5b54567ec0008287d11a2e9e22a

Description

Python ရဲ့ Hashlib ကို အသုံးပြုထားတာဖြစ်ပါတယ်။ ဒီ Module က cryptography ပညာရပ်နဲ့ သက်ဆိုင်တဲ့ Module တစ်ခု ပဲ ဖြစ်ပါတယ်။ Security ပိုင်းဆိုင်ရာ နဲ့ Hash algorithm ပိုင်းမှာအဓိက အသုံးပြုထားတယ်။ ယခုရေးထားတဲ့ပုံစံ က user ထည့်လိုက်တဲ့ password or str data type ကို hash အဖြစ်ပြောင်းလိုက်တာဖြစ်ပါတယ်။

S-13

```
from datetime import datetime
birth_date='2000-02-20'
date_convert= datetime.strptime(birth_date, "%Y-%m-%d").date()
today = datetime.now().date()
age=today.year-date_convert.year
print(age)
```

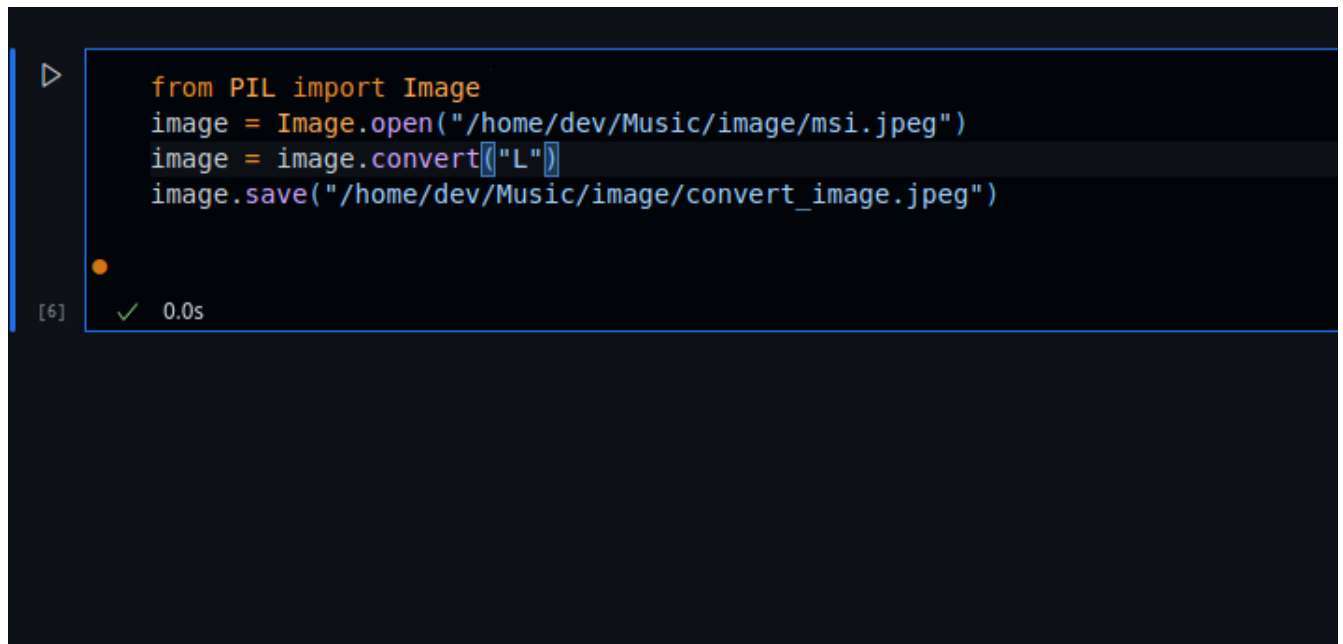
[16] ✓ 0.0s

... 23

Description

Age calculation ဖြစ်တဲ့အတွက် Datetime Module သုံးကိုသုံးရမယ်။ Line 2 မှာ User birthday ကို အတည်ထားလိုက်တာဖြစ်ပါတယ်။ ဒါက string datatype အနေနဲ့တည်ရှိနေတာကြောင့် calculate လုပ်လို့မရဘူး။ အဲ့တာကို Line 3 မှာ Date format အဖြစ်ပြန်ပြောင်းလိုက်တာဖြစ်ပါတယ်။ ပြီးနောက် Line 4 today date ကို သတ်မှတ်လိုက်တာဖြစ်ပါတယ်။ Line 5 မှာ date နှစ်ခုရဲ့ Year ကိုသာ ယူလိုက်ပြီး calculate လုပ်လိုက်တာဖြစ်ပါတယ်။ Line 6 က တော့ရလာတဲ့ result ကို output ထုတ်ပြတာဖြစ်ပါတယ်။

S-14



```
from PIL import Image
image = Image.open("/home/dev/Music/image/msi.jpeg")
image = image.convert("L")
image.save("/home/dev/Music/image/convert_image.jpeg")
```

[6] ✓ 0.0s

Description

Pillow Module ကို အသုံးပြုထားပါတယ်။ second line မှာ ကိုယ်ပြောင်းချင်တဲ့ image ရဲ့ file path ကို ထည့်ပေးရပါတယ်။ နောက်တစ်ကြောင်းက အဲ့ image ကို convert လိုက်တာဖြစ်ပါတယ်။နောက်ဆုံးအကြောင်း ကတော့ convert ပြောင်းလဲတဲ့ image file path နေရာကို သတ်မှတ်ပေးထားလိုက်တာဖြစ်ပါတယ်။ဒီနေရာမှာတစ်ခုက ချက်ချင်းတော့ image ပုံမပြောင်းသေးပဲ base 64 file အနေနဲ့ save ပြီး ခန့် အကြာမှာ image ပုံလေးကိုပြောင်းသွားမှာဖြစ်ပါတယ်။

S-15

```
... 'Linux'

import socket
s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
s.connect(("8.8.8.8", 80))
ip_address = s.getsockname()[0]
print("LAN IP address:", ip_address)

[25] ✓ 0.0s

... LAN IP address: 192.168.1.103
```

Description

Networking နဲ့ ပတ်သတ်တဲ့အပိုင်းဖြစ်တဲ့အတွက် socket module ကိုအသုံးပြုထားပါတယ်။ Line 2 ကတော့ object တစ်ခုကိုတည်ဆောက်တာဖြစ်ပြီး။ Line 3 ကတော့ Google ရဲ့ DNS system ကိုလှမ်းချိတ်လိုက်တာ ဖြစ်ပါတယ်။ Line 4 ကတော့ ထိုချိတ်ဆက်ပြီးမှ return ပြန်လာတဲ့ ip and port ထဲက Ip address ကိုယူလိုက်တာ ဖြစ်ပါတယ်။ Line 5 ကတော့ Output ထုတ်ပြထားတာဖြစ်ပါတယ်။

S-16

```
import pyautogui
import time
time.sleep(2)
pyautogui.typewrite("Script By A-Coder")

✓ 2.1s

Script By A-Coder

Hte Arkar
```

Description

Pyautogui and time module ကို အသုံးပြုထားတာဖြစ်ပြီး အချိန် ၂ စက္ကန့်ကြာပြီးနောက် cursor ထောက် လိုက်တဲ့ နေရာမှာ Script By A-Coder ဆိုတဲ့ စာသားကို auto ရိုက်ပေးရန်အတွက် typewrite function ကိုအသုံးပြု ထားပါတယ်။

S-17



```
import phonenumbers as pn
from phonenumbers import geocoder
ph_number=input("Enter Phone number with Country code: ")
ph_number=pn.parse(ph_number)
print(f'{ph_number} is from {geocoder.description_for_number(ph_number,"en")}')

[10] ✓ 8.1s Python
... Country Code: 95 National Number: 9778497416 is from Myanmar
```

Description

ဒီ Challenge မှာတော့ phonenumbers ဆိုတဲ့ module ကို အသုံးပြုထားပါတယ်။ ပြီးရင် country ဆွဲထုတ် ဖို့ geocoder ကိုပါ import လုပ်ထားပါတယ်။ Line 1 and Line 2 အရ လိုအပ်တဲ့ Module တွေကို import လုပ်ထား တာဖြစ်ပါတယ်။ ထို့နောက် user ဆီက phone number ကို တောင်းပါတယ်။ ဒီနေရာမှာ သတိထားဖို့က +95 ဘာညာ country code လေးရှေ့ကနေ ထည့်ပေးဖို့လိုတာပေါ့ဗျ။ အောက်ကတော့ ph num ကို parse လုပ်ပြီး အောက်ဆုံး ကတော့ output ထုတ်ပြတာဖြစ်ပါတယ်။

S-18



```
user_info=[{'name':'Htet Arkar','Phone':'+959778497416','email':'mamakaunglay@gmail.com'},
            {'name':'Phyo Myat Han','Phone':'+9593232423','email':'sormashikaunglay@gmail.com'}]
user_name=input("Enter Your Name: ")
for user in user_info:
    if user['name']==user_name:
        print(user)
        break

[12] ✓ 7.8s Python
... {'name': 'Phyo Myat Han', 'Phone': '+9593232423', 'email': 'sormashikaunglay@gmail.com'}
```

Description

User Info တွေပါတဲ့ Dictionary List တစ်ခု တည်ဆောက်လိုက်တယ်။ ပြီးနောက် User ကို သူ့ရှာချင်တဲ့ နာမည် ထည့်ခိုင်းတယ်။ အဲ့နောက် Dictionary List ကို Loop ပတ်ပြီး user ရှာလိုက်တဲ့ကောင်ပါရင် loop ရပ်လိုက် ပြီး သက်ဆိုင်ရာ Data ကို ပြပေးပါတယ်။

S-19

```
from bs4 import BeautifulSoup
import requests
url = "https://www.facebook.com"
req = requests.get(url)
soup = BeautifulSoup(req.text, "html.parser")
company=soup.title
print(company.string)
```

[18] ✓ 0.8s Python

... Facebook - လော့ဂ်အင်ဝင်ရန် (သို့) မှတ်ပုံတင်ရန်

Description

ခု challenges က တော့ web script challenges လေးတစ်ခုပဲဖြစ်ပါတယ်။ နောက်ကျ ထပ်ပါလာ အုံးမှာပါ။ bs4 and requests module နှစ်ခုကို အသုံးပြုထားပါတယ်။ bs4 ကိုကျတော့ website ထဲက data တွေကို လက်ခံ ပိုင်ခြား ဘာညာ သရကာ အတွက် အသုံးပြုထားဖြစ်ပါတယ်။ Data Science အတွက်လဲ အသုံးဝင်တဲ့ module တစ်ခုဖြစ်ပါတယ်။ requests က သိတဲ့အတိုင်းလေ့လာကြည့်ပေါ့။ Program ကတော့ ရိုးရှင်းပါတယ်။ facebook site ထဲက html code တွေကို requests က ဖမ်းယူလိုက်တော့ ပြီးနောက် bs4 ထဲက function တွေနဲ့ လိုချင်တဲ့ data ကိုပဲ output ထုတ်လိုက်တာဖြစ်ပါတယ်။

Question or Challenge

Q-20.

URL parameter ရဲ့ value ကို ယူပြပါ။

Q-21.

Program ကို run တဲ့အခါ normal user ကို run ခွင့်ပြီး၊ root user ဆိုရင် It Dangerous ဆိုပြီး run ခွင့်မပေးပါနဲ့။(Linux user)

Q-22.

Postgresql server ကို ချိတ်ပြပါ။

Q-23.

Date ကို ပြ ထားတဲ့ format အတိုင်း output ထုတ်ပြပါ။

Q-24.

Base64 converter tool တစ်ခုရေးပါ။ (ကိုယ်က စာသားထည့်လိုက်ရင် ထိုစာသားကို base64 format ပြောင်းပေးရပါမည်။)

Q-25.

Excel file တစ်ခု create လုပ်ပြီး၊ ထို file ထဲကို data တချို့ထည့်ထားပါစေ။

Q-26.

Site တစ်ခု ရဲ့ HTML source code ကို output ထုတ်ပြပါ။

Q-27.

Text file or document file တစ်ခုခု ကို PDF file အဖြစ်ပြောင်းပေးပါ။

Q-28.

Database table ထဲကို data အသစ်တွေထည့်ပြပါ။

Solution

S-20

```
from urllib.parse import urlparse, parse_qs
url = '127.0.0.1/store?writer=acoder&script=python'
parse_result = urlparse(url)
get_parameter = parse_qs(parse_result.query)
for value in get_parameter:
    print(value,'=>',get_parameter[value])
```

[6] ✓ 0.0s

```
... writer => ['acoder']
    script => ['python']
```

Description

Python ရဲ့ URL lib module ကို အသုံးပြုထားတာဖြစ်ပါတယ်။ ဒီနေရာ မှာ URL ရဲ့ Method နှစ်ခုကို အသုံးပြုထားပါတယ်။ urlparse နဲ့ parse_qs နှစ်ခုပဲဖြစ်ပါတယ်။ Urlparse method ကို URL parsing လုပ်ဖို့ အတွက်အသုံးပြုပါတယ်။ parse_qs က URL parameters တွေကို dict datatype အဖြစ်ပြန်ထုတ်တဲ့နေရာမှာ အသုံးပြုပါတယ်။ Line 3,4 နဲ့က URL Lib ရဲ့ method တွေကိုအသုံးပြုပြီး အောက်ဆုံး for loop ကရလာတဲ့ dictionary datatype ကို line by line ဖြစ် output ထုတ်ပြတာဖြစ်ပါတယ်။

S-21

```
import os
import pwd
linux_user=pwd.getpwuid(os.getuid())[0]
if linux_user == 'root':
    print('Not Access to run in root user')
else:
    print('success',linux_user)
print(linux_user)
```

[7] ✓ 0.0s

```
... success dev
    dev
```


Description

ဒီ Challenge မှာတော့ linux ရဲ့ usertype ကို check ထားပြီး run ခွင့်ပေးမပေး ဆုံးဖြတ်ထားတာဖြစ်ပါတယ်။ OS နဲ့ pwd module နှစ်ခု အသုံးပြုထားတာဖြစ်ပါတယ်။ os module က system နဲ့တိုက်ရိုက်ချိတ်ဆက် အသုံးပြုပြီး၊ pwd ကတော့ Linux environment နဲ့တိုက်ရိုက်ချိတ်ဆက် အသုံးပြုပါတယ်။ အဲနှစ်ခုကိုအသုံးပြုပြီး root ဆို run ခွင့်မပေးဘူး၊ root မဟုတ်ရင် run ခွင့်ပေးမယ်ဆိုပြီး သတ်မှတ်ထားတာဖြစ်ပါတယ်။ ကိုယ်တိုင် run ကြည့်မှ နားလည်ပါလိမ့်မယ်။

S-22

```
import psycpg2
database_connect = psycpg2.connect(
    database = 'stu',
    host = 'localhost',
    port = '5432'
)
if database_connect:
    print("success")
else:
    print("Not Success")
```

[1] ✓ 0.2s Python

... success

Description

Psycpg2 module ကိုအသုံးပြုပြီး database connect လုပ်ပြထားတာဖြစ်ပါတယ်။ connect ဖြစ်လို့ အောင်မြင် မမြင်ပါ connection စစ်ပြထားသေးပါတယ်။

S-23

```
from datetime import datetime
import calendar
today = datetime.now()
print(today.day, calendar.month_name[today.month], today.year)
```

[10] ✓ 0.0s

... 2 September 2023

Description

ဒီ ပိုင်းမှာ datetime နဲ့ calendar module နှစ်ခုသုံးထားပါတယ်။ Day Month Year ကိုယူဖို့အတွက် အသုံးပြုထားပါတယ်။ Calendar Module ကိုကျတော့ Month ရဲ့ Name ကို ယူဖို့အတွက်အသုံးပြုထားပါတယ်။

S-24

```
import base64
text_input=input("Enter Text")
base64.b64encode(text_input.encode('utf-8'))

[11] ✓ 4.9s

... b'aGVsbG8='
```

Description

Base64 module ကို အသုံးပြုထားတာဖြစ်ပါတယ်။ User က ထည့်လိုက်တဲ့ text ကို လက်ခံ base64 format ပြန်ပြောင်းထားတာဖြစ်ပါတယ်။

S-25

```
import xlswriter
workbook = xlswriter.Workbook('user_info.xlsx')
worksheet = workbook.add_worksheet()
worksheet.write(0,0,'Name')
worksheet.write(0,1,'Age')
worksheet.write(1,0,"Acoder")
worksheet.write(1,1,"23")
workbook.close()
```

[13] ✓ 0.0s

Description

Python ရဲ့ xlswriter module ကိုအသုံးပြုထားတာဖြစ်ပါတယ်။ သူက Excel နဲ့တွဲဖက်အသုံးပြုလိုရတဲ့ Python module အသုံးပြုထားတာဖြစ်ပါတယ်။ workbook method နဲ့ excel file ကို create လုပ်တာဖြစ်ပါတယ်။ ထို့နောက် worksheet နဲ့ connect လုပ်လိုက်တာဖြစ်ပါတယ်။ write method ထဲ excel file ထဲကို data ထည့်လိုက်တာဖြစ်ပါတယ်။ program run ပြီးနောက် အောက်ပါအတိုင်း output ထွက်လာပါလိမ့်မယ်။

user_info.xlsx

	A	B
1	Name	
2	Acoder	23

သက်ဆိုင်ရာ folder မှာ workbook မှထည့်ထားတဲ့ file နာမည်လေးနဲ့ ပေါ်နေပါလိမ့်မယ်။

S-26

```
import requests as re
red=re.get('https://www.google.com')
print(red.content)
```

✓ 1.4s Python

b'<!doctype html><html itemscope="" itemtype="http://schema.org/WebPage" lang="my"><head><meta content="text,

Description

ဒီ အပိုဒ်ကျတော့ requests module ကို အသုံးပြုထားပါတယ်။ Requests module ကို http method တွေနဲ့ တွဲသုံးပါတယ်။ ဒီမှာကျတော့ get method နဲ့ site ကိုယူထားပြီး သူထဲက content တွေကို output ထုတ်ပြထားတာ ဖြစ်ပါတယ်။

S-27

```
import aspose.words as aw
doc = aw.Document("/home/dev/Downloads/odoo/odoo16/students/my_info.txt")
doc.save("info.pdf")
```

[3] ✓ 1.5s

.. <aspose.words.saving.SaveOutputParameters object at 0x7eff0821c910>

Description

aspose-words module ကို အသုံးပြုထားတာဖြစ်ပါတယ်။
Line 2 : ကျနော်တို့ ပြောင်းမဲ့ file location ကို ထည့်ပေးရတာပါ။
Line 3 : Line 2 က file ကို pdf ပြောင်းလိုက်တာဖြစ်ပါတယ်။

S-28

```
import psycopg2
db_connect=psycopg2.connect(
    host="localhost",
    database="test_db")
db_connect.autocommit = True
cur = db_connect.cursor()
cur.execute("insert into student (student_id,name,address,level_test) values (1,'arkar','Magway','A-1')")
db_connect.commit()
db_connect.close()
```

Description

Postgresql နဲ့ ပဲစမ်းပြထားပါတယ်။ psycopg2 module ကို အသုံးပြုထားပါတယ်။ ပြီးနောက် Database နဲ့ connect လုပ်ပါတယ်။ autocommit false ဖြစ်နေတာ true ပြောင်းပေးထားပါတယ်။ ထို့နောက် sql query run တဲ့ နေရာမှာ အဆင်ပြေဖို့အတွက် cursor တစ်ခုကို create လုပ်ပါတယ်။ ထို့နောက် query run ပြီး commit ပြန်ချ လိုက်ပါတယ်။

Q-29

Website ရဲ့ source code ကို output ထုတ်ပြပါ။

Q-30

Number or string ကို binary ပြောင်းပြပါ။

Q-31

website တစ်ခုတည်းက link တွေကို ဆွဲထုတ်ပြပါ။

Q-32

Year, Number(Income or student) နှစ်ခုနဲ့ Graph တစ်ခုဆွဲပြပါ။

Q-33

Website ကို Access လုပ်မဲ့ ကိုယ်ရဲ့ header Information ကို change ပြပါ။

Q-34

ကိုယ့်ရဲ့ software ဖွင့်နေလား မဖွင့်နေလားဆိုတာ စစ်ဆေးပြပါ။

Q-35

system ကို shutdown ချတဲ့ script ရေးပြပါ။

Q-36

Json file data တွေကို Read လုပ်ပြပါ။

Q-37

Text File ထဲက line တွေကို line number နဲ့ ပြန်ပြီး line by line output ထုတ်ပြပါ။

Q-38

Python နဲ့ website တစ်ခုကို Login ဝင်ပြပါ။

Q-39

Customer ကို အကြွေး limit နဲ့ ဝယ်သုံးခွင့်ပေးထားတယ်။ အဲ့ limit ကျော်တာနဲ့ alert ပြပေးပါ။

Solution

S-29

```
import requests
site = requests.get('https://www.google.com')
print(site.content)
```

[1] ✓ 1.6s

... b'<!doctype html><html itemscope="" itemtype="http://schema.org/WebPage" lang="my"

Description

Python ရဲ့ Requests module ကို အသုံးပြုထားပါတယ်။ Requests module ကို HTTP/HTTPS method တွေနဲ့ တွဲဖက်အသုံးပြုရပါတယ်။ ခုက get အသုံးပြုထားပြီး site ရဲ့ content တွေကို ရယူလိုက်တာဖြစ်ပါတယ်။ ထို့နောက် print function နဲ့ output ထုတ်ပြထားတာဖြစ်ပါတယ်။

S-30

```
','.join(map(bin,bytearray('Hello','utf8')))
```

[3] ✓ 0.0s

... '0b1001000,0b1100101,0b1101100,0b1101100,0b1101111'

Description

Join ကတော့ binary တစ်လုံးပြောင်းပြီးရင် comma နဲ့ ခြားပေးဖို့သုံးပြီး နောက်က တော့ string ကို binary ပြောင်းလိုက်တာဖြစ်ပါတယ်။

```

import requests
from bs4 import BeautifulSoup

reqs = requests.get('https://www.google.com')
soup = BeautifulSoup(reqs.text, 'html.parser')

urls = []
for link in soup.find_all('a'):
    print(link.get('href'))

```

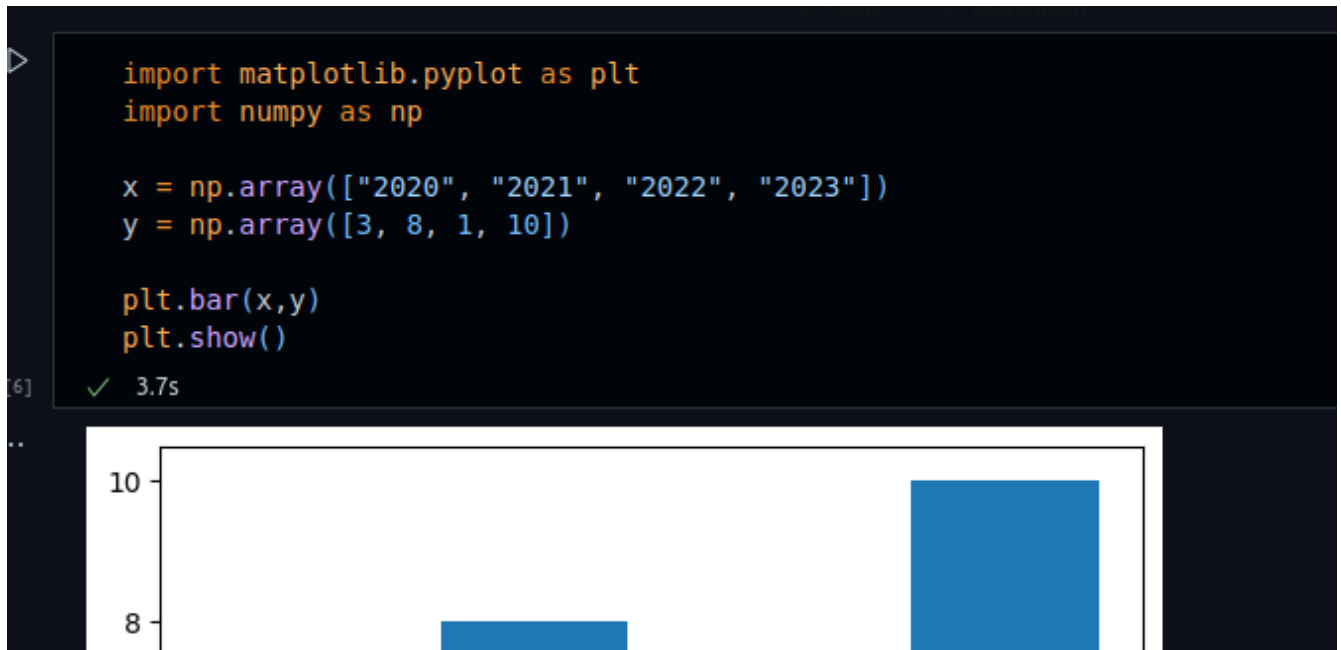
✓ 1.2s

<https://www.google.com/imghp?hl=my&tab=wi>
<https://maps.google.com.mm/maps?hl=my&tab=wl>
<https://play.google.com/?hl=my&tab=w8>
<https://news.google.com/?tab=wn>
<https://mail.google.com/mail/?tab=wm>
<https://drive.google.com/?tab=wo>
<https://calendar.google.com/calendar?tab=wc>

Description

ဒီ မှာဆို BeautifulSoup(bs4) module ကို အသုံးပြုထားပါတယ်။ Bs4 module ကို HTML and XML file ထဲက data ကို ဆွဲထုတ်တဲ့အခါမှာအသုံးပြုပါတယ်။ ဒီမှာဆို requests နဲ့ html ကို ရအောင်ယူတယ်။ ပြီးနောက် bs4 နဲ့ချိတ်ဆက်တယ်။ find_all နဲ့ a tag ကို ဆွဲထုတ်လိုက်တယ်။ အဲ့ထဲကနေ href attribute ပါတဲ့ကောင်ကို ပဲသီးသန့်ဆွဲထုတ်ပြီး output ထုတ်ပြတယ်။

S-32



Description

Matplotlib နဲ့ Numpy module နှစ်ခုကို အသုံးပြုထားပါတယ်။ Numpy module က array နဲ့ တွဲပြီး အသုံးပြု ပါတယ်။ Matplotlib module ကတော့ အဲ array တွေကို graph တစ်ခု အဖြစ် output ထုတ်တဲ့နေရာမှာအသုံးပြုပါတယ်။ဒီမှာ ဆို bar အဖြစ် output ထုတ်ပြထားပါတယ်။

S-33

```
import requests as req
response = req.post('https://www.google.com', headers={'User-Agent': 'Mozilla/5.0 (Macintosh;)'})
print(response.headers)
```

✓ 0.3s

{'Allow': 'GET, HEAD', 'Date': 'Thu, 21 Sep 2023 07:10:26 GMT', 'Content-Type': 'text/html; charset=

Description

Python ရဲ့ requests module ကို အသုံးပြုထားပါတယ်။ header က website ကို access လုပ်မဲ့ header အချက်အလက် ကို json format(object) ဖြင့် ထည့်ပေးလိုက်ပါတယ်။ headers နဲ့နေရာမှာ data ဆိုပြီး website login ဝင်ကြည့်လို့ရပါတယ်။

S-34

```
import psutil

# check if chrome is open
"chrome" in (i.name() for i in psutil.process_iter())
✓ 0.0s

True
```

Description

Psutil module ကို အသုံးပြုထားပါတယ်။ Psutil module ကို system monitoring and system utilization လုပ်တဲ့နေရာမှာအသုံးပြုပါတယ်။ ခု program မှာ running လုပ်နေတဲ့ process တွေထဲမှာ chrome ကို ဖွင့်မဖွင့်ထားစစ်ထားပါတယ်။ စစ်ထားရင် True လို့ return ပြန်ပါတယ်။ (Note:ကျနော် linux မို့ chrome လို့ထည့်ထားတာဖြစ်ပြီး၊ windows ဆိုရင်တော့ chrome.exe လို့ထည့်ရပါမယ်။)

S-35

```
import os
os.system('shutdown now')
✗ 0.0s
```

Description

os module ကိုအသုံးပြုထားပါတယ်။ os.system module က terminal ကို အသုံးပြုလိုက်တာနဲ့တူတူပဲဖြစ်ပြီး အဲ့ထဲမှာ ကိုထည့်ချင်တဲ့ command ကိုရိုက်ထည့်လိုက်တာဖြစ်ပါတယ်။ ကျနော်က shutdown now ပေါ့။

S-36

```
import json
data = json.load(open('my_info.json'))
for i in data['my_info']:
    print(i)
✓ 0.0s
{'name': 'acoder', 'book': 'Python Challenges', 'Price': 'Free'}
```

Description

Python ရဲ့ Json module ကို အသုံးပြုထားပါတယ်။ Json.load ကို json မှ python ရဲ့ dictionary data type အဖြစ် return ပြန်ရန်အတွက် အသုံးပြုပါတယ်။ထို့နောက် json data တွေကို output ထုတ်ပြလိုက်ပါတယ်။

S-37

```
file1 = open('example.txt', 'r')
Lines = file1.readlines()
count = 0
for line in Lines:
    count += 1
    print("Line{}: {}".format(count, line.strip()))
✓ 0.0s
Line1: acoder
Line2: book
Line3: dadaf
Line4: diafkjda
Line5: dkafd
Line6: dkaldf
Line7: cldad
Line8: ckdafe
```

Description

File ရဲ့ readline function ကိုအသုံးပြုထားတာဖြစ်ပြီး line 1 ကြောင်း output ထုတ်တိုင်း line number နဲ့ အတူ output ထုတ်ရန်အတွက် format ကို အသုံးပြုထားပါတယ်။

S-38

```
import requests as req
login=req.post('https://www.example.com/login',data={"email":"acoderacoder@gmail.com","password":"password123"})
✓ 1.4s
```

Description

Requests module ကို အသုံးပြုထားတာဖြစ်ပြီး login ဝင်မှာဖြစ်သောကြောင့် post method ကို အသုံးပြုထားတာဖြစ်ပါတယ်။ ထို့နောက် data တွေထည့်ပြီး run လိုက်ပါတယ်။

S-39

```
limit_amount=30000
customer_sale=input("Total Amount:")
if int(customer_sale) > limit_amount :
    print("Limited Amount")
✓ 4.1s
```

Limited Amount

Description

Limit Amount သတ်မှတ်ထားတာဖြစ်ပြီး ထို limit amount ထည့်ကျော်လွန်းသွားပါက ထက်ပြီး ဝယ်ခွင့်မရှိပါ။

Q-40

လက်ရှိ battery အားကို output ထုတ်ပြပါ။

Q-41

System ထဲမှာ run နေတဲ့ process တစ်ခုကို kill ပြပါ။

Q-42

LAN ထဲမှာ Run နေတဲ့ IP address တွေကို output ထုတ်ပြပါ။

Q-43

အသုံးပြုနေတဲ့ Memory amount ကို output ထုတ်ပြပါ။

Q-44

အသုံးပြုနေတဲ့ CPU amount ကို output ထုတ်ပြပါ။

Q-45

ကိုယ် system ရဲ့ အချက်အလက်တွေကို output ထုတ်ပြပါ။

Q-46

ကိုယ်စက်ရဲ့ network info ကို output ထုတ်ပြပါ။

Q-47

Python နဲ့ ssh connect လုပ်ပြပါ။

Q-48

အနီးနားရှိ Bluetooth device များကိုရှာပြပါ။

Q-49

ကိုယ်ရဲ့ HDD storage amount ကို output ထုတ်ပြပါ။

Q-50

Discover available wifi network.

Solution

S-40

```
import psutil

battery_status = psutil.sensors_battery()
print("Percentage of battery",battery_status.percent)
```

✓ 0.0s

Percentage of battery 35.294117647058826

Description

psutil module ကို အသုံးပြုထားတာဖြစ်ပြီး system ရဲ့ battery နဲ့ ချိတ်ဆက်ထားလိုက်တာဖြစ်ပါတယ်။ ထို့နောက် percent method ကို အသုံးပြုပြီး လက်ရှိ battery ရဲ့ percent ကို အသုံးပြုထားပါတယ်။

S-41

```
import os
os.system('kill -9 3444')
```

✓ 0.0s

Description

os module ကို အသုံးပြုထားတာဖြစ်ပါတယ်။ထုံစံအတိုင်း linux ရဲ့ process kill command ကို အသုံးပြုထားပြီး process ကို kill လိုက်တာဖြစ်ပါတယ်။

S-42

```
import netifaces
lan_data = netifaces.ifaddresses('wlp2s0')
for i in lan_data:
    for data in lan_data[i]:
        for ip in data:
            print("IP ADDR in LAN", "=>", data['addr'])
```

6] ✓ 0.0s

```
IP ADDR in LAN => a0:51:0b:95:7b:cb
IP ADDR in LAN => a0:51:0b:95:7b:cb
IP ADDR in LAN => 192.168.1.131
IP ADDR in LAN => 192.168.1.131
IP ADDR in LAN => 192.168.1.131
IP ADDR in LAN => fd91:2ae4:f95f:493d:132d:842d:3d74:3a8d
IP ADDR in LAN => fd91:2ae4:f95f:493d:132d:842d:3d74:3a8d
IP ADDR in LAN => fd91:2ae4:f95f:493d:d773:9726:8822:9632
IP ADDR in LAN => fd91:2ae4:f95f:493d:d773:9726:8822:9632
IP ADDR in LAN => fe80::619c:88d6:9394:a53b%wlp2s0
```

Description

Netifaces module ကို အသုံးပြုထားတာဖြစ်ပါတယ်။ဒီ module ကို LAN အတွင်းရှိ device တွေရဲ့ name,mac address ကို စစ်ထုတ်တဲ့အခါမှာအသုံးပြုပါတယ်။

Line 1: module ကို import လုပ်ထားတာဖြစ်ပါတယ်။

Line 2: WLAN နဲ့ချိတ်ဆက်လိုက်တာဖြစ်ပါတယ်။

Line 3: ချိတ်ဆက်ထားတဲ့ ကောင်နဲ့ပတ်သတ်တဲ့ IP,Mac and etc စသည့်တို့ကို loop ပတ်ထားတာဖြစ်ပါတယ်။

Line 4: json format နဲ့ output ထုတ်ပြတာဖြစ်သောကြောင့် key ပါတဲ့ ကောင်တွေရဲ့ data တွေကို တစ်ကြောင်းချင်း ထွက်ရန် Loop ပတ်ထားတာဖြစ်ပါတယ်။

Line 5 and 6 : အဲ့ထဲကမှ IP address ကို ပဲသီးသန့်ထုတ်ပြလိုက်တာဖြစ်ပါတယ်။

S-43

```
▶ import psutil
  print(psutil.virtual_memory().percent)
[29] ✓ 0.0s
... 37.5
```

Description

psutil module ကို အသုံးပြုထားတာဖြစ်ပါတယ်။ virtutual_memory().percent ဆိုတာ က ကျနော်လက်ရှိ အသုံးပြုထားတဲ့ run ရဲ့ ပမာဏ ကို percent ဖြင့် output ထုတ်ပြရန်အတွက်ဖြစ်ပါတယ်။

S-44

```
▶ import psutil
  print(psutil.cpu_percent())
[38] ✓ 0.0s
... 10.7
```

Description

CPU ရဲ့ အသုံးပြုထားတဲ့ ပမာဏ ကို output ထုတ်ပြထားတဲ့ program တစ်ခုဖြစ်ပါတယ်။

S-45

```
import platform
print('Operating system:', platform.system())
print('Host Name:', platform.node())
print('processor:', platform.processor())
print('Architecture:', platform.architecture())
```

✓ 0.0s

```
Operating system: Linux
Host Name: dev
processor: x86_64
Architecture: ('64bit', 'ELF')
```

Description

Platform module ကို အသုံးပြုထားတာဖြစ်ပြီး အသုံးပြုနေတဲ့ os name၊ host name နဲ့ processor ကို output ထုတ်ပြထားတဲ့ program ဖြစ်ပါတယ်။ တခြားသော အချက်အလက်များကို စစ်ထုတ်ကြည့်လိုပါက platform module ကိုလေ့လာနိုင်ပါတယ်။

S-46


```
import ifcfg
my_network = ifcfg.interfaces()
for details in my_network:
    print(details,'=>',my_network[details])
```

✓ 0.0s

br-30f4d123ef99 => {'inet': '172.18.0.1', 'inet4': ['172.18.0.1'], 'ether': '02:42:45:87:0a:9b', 'inet6':
docker0 => {'inet': '172.17.0.1', 'inet4': ['172.17.0.1'], 'ether': '02:42:ad:5f:53:bd', 'inet6': [], 'ne
enp1s0 => {'inet': None, 'inet4': [], 'ether': 'e4:e7:49:41:00:c2', 'inet6': [], 'netmask': None, 'netmas
lo => {'inet': '127.0.0.1', 'inet4': ['127.0.0.1'], 'ether': None, 'inet6': ['::1'], 'netmask': '255.0.0.
vmnet1 => {'inet': '172.16.91.1', 'inet4': ['172.16.91.1'], 'ether': '00:50:56:c0:00:01', 'inet6': ['fe80
vmnet8 => {'inet': '172.16.31.1', 'inet4': ['172.16.31.1'], 'ether': '00:50:56:c0:00:08', 'inet6': ['fe80
wlp2s0 => {'inet': '192.168.100.63', 'inet4': ['192.168.100.63'], 'ether': 'a0:51:0b:95:7b:cb', 'inet6':

Description

ဒီတစ်ခါတော့ ifcfg module ကိုအသုံးပြုထားတာဖြစ်ပြီး ကျနော် network ရဲ့ info တော်တော်များများကို output ထုတ်ပြပါ။ Program ကို တော့နားလည်မယ်လို့ ယူဆပါတယ်။ S-42 နဲ့ဆင်လို့ try ကြည့်ပါ။

S-47

```
import paramiko
ssh = paramiko.SSHClient()
ssh.connect("172.23.3.4",
            username="admin",
            password="admin",
            look_for_keys=False )
```

✗ 9.3s

Description

Paramiko module ကို အသုံးပြုထားပါတယ်။ ssh နဲ့ဆိုင်တဲ့ module ဖြစ်ပြီး client ရော server မှာပါ အသုံးပြုလို့ရပါတယ်။ ခု program ကတော့ server ကို connect လုပ်ပြတဲ့ program အသေးစားလေးတစ်ခုဖြစ်ပါတယ်။

S-48

```
import bluetooth
nearby_devices = bluetooth.discover_devices(lookup_names=True)
for addr, name in nearby_devices:
    print(" {} - {}".format(addr, name))
```

✓ 10.9s

BC:2D:EF:DD:53:CE - realme C11

Description

PyBluez2 module ကို အသုံးပြုထားတာဖြစ်ပါသည်။ ဒီ module က Bluetooth နဲ့ သက်ဆိုင်တဲ့ resource တွေကို access ရယူတဲ့ နေရာမှာဖြစ်ပါတယ်။ discover_devices method ကို အသုံးပြုထားတာဖြစ်ပြီး အနီးရှိပွင့်နေတဲ့ Bluetooth နဲ့ name ကို output ထုတ်ပြထားပါတယ်။

S-49

```
import psutil
print("My HDD",psutil.disk_usage('/'))
print('My HDD',psutil.disk_partitions())
```

✓ 0.0s Python

My HDD sdiskusage(total=490577010688, used=360584974336, free=104996818944, percent=77.4)

My HDD [sdiskpart(device='/dev/sda2', mountpoint='/', fstype='ext4', opts='rw,relatime,errors=remount-ro', m

Description

psutil module ကို အသုံးပြုထားတာဖြစ်ပါတယ်။ disk_usage and disk_partitions method နှစ်ခုကို အသုံးပြုထားတာဖြစ်ပြီး ပထမ method ကတော့ total size, free size, used size တို့ကို output ထုတ်ပြတာဖြစ်ပြီး၊ ဒုတိယ တစ်ခုကတော့ partitions ကို output ထုတ်ပြထားတာဖြစ်ပါတယ်။

S-50

```
import subprocess
devices = subprocess.check_output(['nmcli','dev','wifi'])# displaying the information
print(devices)
```

✓ 1.0s Python

IN-USE	BSSID	SSID	MODE	CHAN	RATE	SIGNAL	BARS	SECURITY	\n
--------	-------	------	------	------	------	--------	------	----------	----

Description

Subprocess module ကို အသုံးပြုထားပါတယ်။ ဒီ module က program ကို process တစ်ခုအဖြစ် create လုပ်ပြီး run ပါတယ်။ check_output function ကို အသုံးပြုပြီး အနားရှိ available wifi တွေကို scan လုပ်ပြထားတာ ပါတယ်။

Q-51

Tables တစ်ခုကို postgresql Database တစ်ခုတည်းမှာ ထဲမှာတည်ဆောက်ပြပါ။

Q-52

51 ထဲဆောက်ထားတဲ့ Table ထဲကို Data ထည့်ပြပါ။

Q-53

51 ထဲက table ထဲက data တွေကို output ထုတ်ပြပါ။

Q-54

51 table ထဲက user input ထည့်လိုက်တဲ့ data တစ်ကြောင်းကို update လုပ်ပေးပါ။

Q-55

User ထည့်လိုက်တဲ့ condition ရှိတဲ့ data ကို ဖျက်ပေးပါ။

Q-56

Postgresql ထဲက Data တွေကို Excel file ဖြင့် output ထုတ်ပြပါ။

Q-57

CSV file အား သူ့ Format အတိုင်း output ထုတ်ပြပါ။

Q-58

Data Frame ကို SQL data အဖြစ် create လုပ်ပြပါ။

Q-59

Excel File ထဲက Data တွေကို SQL data အဖြစ် insert လုပ်ပြပါ။

Q-60

PDF တစ်ခုကို python ဖြင့် create လုပ်ပါ။

Solution

S-51

```
+ Code + Markdown | ▶ Run All ↺ Restart | ▶ Execute Group 1 || ▶ Execute C

▷ v

import psycopg2 as pg
db = pg.connect(
    database = 'lesson',
    host = 'localhost',
    port = '5432'
)
db.autocommit = True
cur = db.cursor()
cur.execute("""
CREATE TABLE grades(
    name VARCHAR ( 50 ),
    class_no INT
)
""")
db.commit()
db.close()

[33] ✓ 0.3s
```

Description

psycopg2 module ကိုအသုံးပြုထားတာဖြစ်ပါတယ်။ connect method နဲ့ postgresql server ကို connect လုပ်လိုက်တာဖြစ်ပါတယ်။ ထို့နောက် cursor တစ်ခုကို create လုပ်လိုက်တာဖြစ်ပြီး execute နဲ့ sql query ကို run လိုက်တာဖြစ်ပါတယ်။

S-52

```
import psycopg2 as pg
db = pg.connect( database = 'lesson', host = 'localhost', port = '5432')
db.autocommit = True
cur = db.cursor()
cur.execute("insert into grades (name,class_no) values ('Grade-A',1)")
db.commit()
db.close()
```

[16] ✓ 0.0s

... None

Description

SQL query နဲ့ ချိတ်ဆက်အသုံးပြုတဲ့ program ပါပဲ။ အဓိက SQL query ကို run လိုက်တာပဲဖြစ်ပါတယ်။ နားလည်ဖို့ဆို database အကြောင်းလေ့ဖို့လိုပါတယ်။

S-53

```
import psycopg2 as pg
db = pg.connect( database = 'lesson', host = 'localhost', port = '5432')
db.autocommit = True
cur = db.cursor()
cur.execute("select * from grades")
print(cur.fetchall())
db.commit()
db.close()
```

✓ 0.0s

[('Grade-A', 1), ('Grade-A', 1), ('Grade-A', 2), ('Grade-A', 3), ('Grade-A',

Description

ကျန်တာထက် ဒီ program မှာ fetchall() ဆိုတဲ့ method လေးကိုသတိထားမိမယ်ထင်ပါတယ်။သူ့ကိုကျနော်တို့ table ထဲက data တွေအကုန်လုံး output ထုတ်ဖို့အတွက်အသုံးပြုပါတယ်။ တကယ်လို့ ပထမဆုံး တစ်ကြောင်းကိုပဲ output ထုတ်ချင်တဲ့ဆိုရင် fetchone ဆိုပြီး အစားထိုးပေးလိုက်ပါ။

S-54

```
import psycopg2 as pg
db = pg.connect( database = 'lesson', host = 'localhost', port = '5432')
db.autocommit = True
cur = db.cursor()
update = int(input("Enter Number"))
cur.execute(f"update grades set class_no=5 where class_no={update}")
db.commit()
db.close()
```

✓ 1.9s

Description

ဒီနေရာအထက်ပါ program တွေ့နဲ့ ကွာခြားနေတာလေးက input function နဲ့ f” “ ပုံစံလေးဖြစ်နေတာလေးပါ။ Input က user ထည့်လိုက် data ပါတဲ့ ကောင်ကို လက်ခံဖို့သုံးပါတယ်။ f” “ ကတော့ variable နဲ့ string ရောသုံးချင်လို့သုံးတာဖြစ်ပါတယ်။ ဒါကြောင့် query ပြန်ထုတ်လိုက်မယ်ဆိုရင် update grades set class_no=5 where class_no=1 (1 ကတော့ user ဆီက data ကို လှမ်းဖမ်းလိုက်တဲ့အခါမှာအသုံးပြုပါတယ်။

S-55

```
import psycopg2 as pg
db = pg.connect( database = 'lesson', host = 'localhost', port = '5432')
db.autocommit = True
cur = db.cursor()
delete = int(input("Enter Number"))
cur.execute(f"delete from grades where class_no={delete}")
db.commit()
db.close()
```

0.0s

Description

Delete query ကိုအသုံးပြုလိုက်တာပဲဖြစ်ပါတယ်။ CRUD ကို တစ်ခုခုရှင်းပြသွားတာဖြစ်ပါတယ်။နားလည်မယ်လို့ယူဆ ပါတယ်။

S-56

```
import psycopg2 as pg
import xlswriter as xl
db = pg.connect( database = 'lesson', host = 'localhost', port = '5432')
workbook = xl.Workbook('data.xlsx')
worksheet = workbook.add_worksheet()

db.autocommit = True
cur = db.cursor()
cur.execute(f"select * from grades")
datas = cur.fetchall()

row = col = 0
for grade, class_No in (datas):
    worksheet.write(row, col, grade)
    worksheet.write(row, col + 1, class_No)
    row += 1

workbook.close()

db.commit()
db.close()
```

✓ 0.0s Python

Description

ဒီမှာဆိုရင်တော့ xlswriter ကို အသုံးပြုထားပါတယ်။ excel ထဲကို data ထဲဖို့အတွက်ပေါ့။ Line 4 and 5 ကတော့ data.xlsx file တစ်ခုကို ဖန်တီးလိုက်တာဖြစ်ပါတယ်။ အဲ့အောက်က 4 ကြောင်းကတော့ SQL query ကို run တာဖြစ်ပါသည်။ ထို့နောက် row and col ကို 0 place နေရာမှာသတ်မှတ်လိုက်ပါ။ ပြီးနောက် Query ထဲက data တွေကို for loop ပတ်ပြီး excel file ထဲ ထည့်တာဖြစ်ပြီး တစ် row ပြီးရင် တစ်ကြောင်းဆင်းအောင် ရေးထားတာဖြစ်ပါတယ်။

S-57

```
import pandas as pd
print(pd.read_excel('data.xls'))
```

✓ 0.0s

	name	class_no
0	Grade-B	1
1	Grade-C	1

Description

Pandas module ကို အသုံးပြုထားတာဖြစ်ပါတယ်။ သူ့ကို DS နဲ့ ပတ်သတ်တဲ့ နေရာတွေမှာ အသုံးပြုပါတယ်။ read_excel function ကတော့ excel ထဲမှာရေးထားတဲ့ format အတိုင်း output ထုတ်ပြန်တာဖြစ်ပါတယ်။

S-58

```
import pandas as pd
from sqlalchemy import create_engine
engine = create_engine('postgresql+psycopg2://postgres@localhost/lesson')
df = pd.DataFrame([{'name': 'Grade B', 'class_no': 3}])
df.to_sql('grades', engine, if_exists='append', index=False)

0.1s
```

Description

Pandas and SQLAlchemy module နှစ်ခုကို အသုံးပြုထားပါတယ်။ ဒီနေရာမှာ ထူးခြားတာက postgresql ကို sqlalchemy module နဲ့ ချိတ်ဆက်အသုံးပြုထားပါတယ်။ အကြောင်းကတော့ psycopg2 နဲ့ အဆင်မပြေနိုင်လို့ပါ။ Line 3 ကတော့ postgresql နဲ့ ချိတ်ဆက်ဖို့ အတွက် engine တစ်ခုကို create လုပ်တာဖြစ်ပါတယ်။ Line 4 ကတော့ Data Frame တစ်ခုကို တည်ဆောက်လိုက်တာပါ။ ထို့နောက် Line 5 မှာ row တစ်ခုအဖြစ် Data ကို create လုပ်လိုက်တာဖြစ်ပါတယ်။

S-59

```
import pandas as pd
from sqlalchemy import create_engine
engine = create_engine('postgresql+psycopg2://postgres@localhost/lesson')
df = pd.read_excel('data.xls')
df.to_sql('grades', engine, if_exists='append', index=False)

✓ 0.0s
```

Description

58 နဲ့ Data Frame သတ်မှတ်တဲ့အကြောင်းပဲ ကွာပါတယ်။ ဒီနေရာမှာ excel နဲ့ ချိတ်ဆက်ပြီး data တွေကို insert လုပ်ထားတာဖြစ်ပါတယ်။

```
from fpdf import FPDF
pdf = FPDF()
pdf.add_page()
pdf.set_font('Arial', 'B', 16)
pdf.cell(10, 10, "PDF Hello")
● pdf.output('tuto1.pdf', 'F')
```

✓ 0.0s

Description

fpdf module ကို အသုံးပြုထားပါတယ်။ simple pdf create လုပ်တဲ့ program ဖြစ်ပါတယ်။

Q-61

Website ရဲ့ header information တွေကို output ထုတ်ပြပါ။

Q-62

Website ထဲက contact information ကိုပဲသီးသန့် output ထုတ်ပြပါ။

Q-63

Web Domain နဲ့ဆိုင်တဲ့ information တွေကို output ထုတ်ပြပါ။

Q-64

Website ထဲမှာရှိတဲ့ image link တွေကို extract လုပ်ပြပါ။

Q-65

လက်ရှိရဲ့ stock ဈေးကွက်ထဲ usd/jpy currency amount ကို output ထုတ်ပြပါ။

Q-66

Website ရဲ့ current status ကို output ထုတ်ပြပါ။

Q-67

Web Page ရဲ့ အလုပ်လုပ်နေတဲ့ URL path တွေကို output ထုတ်ပြပါ။

Q-68

script ကို run လိုက်တာနဲ့ သတ်မှတ်ထားတဲ့ URL ကို chrome မှာဖွင့်ပေးမှာ။

Q-69

Website ရဲ့ IP address ကို စစ်ထုတ်ပြပါ။

Q-70

ကိုယ့် access လုပ်မဲ့ web page ရဲ့ HTML content ကို file ထဲမှာ save ပြပါ။

Solution

S-61

```
import requests as req
header = req.get('https://www.google.com')
● print(header.headers)
✓ 0.9s
```

```
{'Date': 'Sun, 24 Sep 2023 05:00:53 GMT', 'Expires': '-1', 'Cache-Control'
```

Description

Python ရဲ့ Requests module ကို အသုံးပြုထားတာဖြစ်ပါတယ်။ Get method ကို အသုံးပြုပြီး website ကို access ယူလိုက်ပါတယ်။ အဲဒီမှာ web ရဲ့ headers အချက်အလက်တွေကို output ထုတ်ပြထားတာဖြစ်ပါတယ်။

S-62

```
import requests as req
from bs4 import BeautifulSoup
contact = req.get('https://www.studytonight.com/contact/')
soup = BeautifulSoup(contact.content, "html.parser")
results = soup.findAll("address")
print(results)
```

✓ 0.8s

```
[<address>
<strong>Studytonight Technologies Pvt Ltd</strong><br>
    Co-Offiz, Magnum Tower 1, 8th Floor,<br>
    Golf Course Extension Road,<br>
    Sector 58, Gurugram
</address>]
```

Description

BS4 နဲ့ requests module ကို အသုံးပြုထားတာဖြစ်ပါတယ်။ Line 1 and 2 ကတော့ import လုပ်ခြင်း ဖြစ်သည်။ Line 3 ကတော့ html page content ကိုရယူရန်အတွက် get method နဲ့သုံးထားခြင်းဖြစ်ပြီး Line 4 ကတော့ အဲ့ content ထဲက tag, class and id စသည့်တို့ကိုရှာရန် bs4 နဲ့ ချိတ်ဆက်ခြင်းဖြစ်သည်။အောက်ဆုံး ကတော့ address ဆိုတဲ့ tag နဲ့ ရှာလိုက်ပြီး address ကို scrap လုပ်လိုက်ခြင်းဖြစ်သည်။

S-63

```
> import whois
● print(whois.whois('https://www.example.com'))
[8] ✓ 4.9s
.. {
  "domain_name": "EXAMPLE.COM",
  "registrar": "RESERVED-Internet Assigned Numbers Authority",
  "whois_server": "whois.iana.org",
  "referral_url": null,
  "updated_date": "2023-08-14 07:01:38",
  "creation_date": "1995-08-14 04:00:00",
  "expiration_date": "2024-08-13 04:00:00",
  "name_servers": [
    "A.IANA-SERVERS.NET",
    "B.IANA-SERVERS.NET"
```

Description

Python ရဲ့ whois module အသုံးပြုထားတာဖြစ်ပါတယ်။ whois.whois function နဲ့ domain နဲ့ဆိုင်ရာ အချက်အလက် တွေကို output ပြန်ထုတ်ပေးပါတယ်။

S-64

```
> import requests as req
from bs4 import BeautifulSoup
contact = req.get('https://www.google.com')
soup = BeautifulSoup(contact.content, "html.parser")
for link in soup.find_all('img'):
    print(link.get('src'))
[9] ✓ 2.4s
.. /images/branding/googlelogo/1x/googlelogo\_white\_background\_color\_272x92dp.png
```

Description

Extract All link program နဲ့ဆောင်ပါတယ်။ဒီမှာ ဆို img tag ပါတဲ့ကောင်တွေကိုရှာတယ်။ အဲ့ထဲကမှာ src attribute ပါတဲ့ကောင်တွေကို သီးသန့်ပြန် ပြီး extract လုပ်ပြီး output ထုတ်ပြပါတယ်။

S-65

```
import requests as req
from bs4 import BeautifulSoup
contact = req.get('https://www.google.com/finance/quote/USD-JPY')
soup = BeautifulSoup(contact.content, "html.parser")
print(soup.find("div", class_ = "YMlKec fxKbKc").contents)
```

✓ 3.3s

['148.2800']

Description

Google Live ရဲ့ data ကိုဆွဲယူပြထားပါတယ်။အထူးတလည်တော့ရှင်းပြမနေတော့ပါ။ 62 နဲ့တူတူနည်းပါပဲ ဒီနေရာမှာ သတိထားရမှာက class မဟုတ်ဘူးနော် class_ .

S-66

```
import requests as req
site = req.get('https://www.google.com')
print(site.status_code)
```

✓ 0.5s

200

Description

status_code ကို အသုံးပြုပြီး စစ်ထုတ်ထားပါတယ်။ return ပြန်လာတဲ့ HTTPS status code က integer value ဆိုတာကိုတော့ သတိထားပေးပါ။

S-67

```
import requests as req
list_dir = ['', 'admin', 'img', 'login', 'video', 'aud']
for url in list_dir:
    target = 'https://www.example.com'+ url
    test = req.get(target)
    if test.status_code == 200:
        print(test, target)
    else:
        pass
```

Description

URL path ပါမဲ့ကောင်တွေကို list တစ်ခုကို တည်ဆောက်ထားလိုက်ပါတယ်။ ထို့နောက် for loop ပတ်ပြီး အကောင်တွေနဲ့ target url နဲ့ ပေါင်းပါတယ်။ အဲ့ထဲကနေ status code 200 ရှိတဲ့ကောင်တွေကိုပဲ သီးသန့်ဆွဲထုတ်လိုက်ပါတယ်။

S-68

```
import requests as req
list_dir = ['', 'admin', 'img', 'login', 'video', 'aud']
for url in list_dir:
    target = 'https://www.example.com'+ url
    test = req.get(target)
    if test.status_code == 200:
        print(test, target)
    else:
        pass
```

Description

selenium framework ကို အသုံးပြုထားတာဖြစ်ပါတယ်။ အဲ့ထဲကနေ chrome driver ကို ချိတ်ဆက်ထားပါတယ်။ .get က ဒီတစ်ခါမှာတော့ URL ကို browser မှာဖွင့်ဖို့အတွက်သုံးပါတယ်။

S-69

```
import socket
socket.gethostbyname('google.com')
✓ 0.0s
'64.233.170.102'
```

Description

socket module ကို အသုံးပြုထားတာဖြစ်ပါတယ်။ gethostbyname method ကို အသုံးပြုပြီး ထည့်ထားတဲ့ domain name ကို IP address အဖြစ်ပြောင်းပေးပါတယ်။

S-70

```
import urllib
access_url = urllib.request.urlopen('https://www.google.com')
save_file = open('target.txt','w')
save_file.write(str(access_url.read()))
save_file.close()
✓ 0.5s
```

Description

Python ရဲ့ urllib module ကို အသုံးပြုထားပါတယ်။ Line 2 ကတော့ url ကို access လုပ်ထားတာဖြစ်ပြီး Line 3 က ထွက်လာတဲ့ html ကို save ဖို့အတွက် file name သတ်မှတ်ထားတာဖြစ်ပါတယ်။ အောက် code တွေကတော့ html code တွေကို file ထဲ မှာ save လိုက်တာဖြစ်ပါတယ်။

Q-71

Number Amount ကို word အဖြစ် ပြောင်းပြီး output ထုတ်ပေးပါ။

Q-72

PDF file ရဲ့ meta data ကို output ထုတ်ပြပါ။

Q-73

User နိမ့်လိုက်တဲ့ key ကို track လိုက်ပြပါ။

Q-74

Zip file ကို extract ဖြည့်လိုက်ပါ။

Q-75

Python nmap ဖြင့် simple scanner တစ်ခု create လုပ်ပြပါ။

S-71

```
> import inflect
p = inflect.engine()
p.number_to_words(150000)
[51] ✓ 0.1s
... 'one hundred and fifty thousand'
```

Description

inflect module ကို အသုံးပြုပြီး ပြောင်းပြောင်းပါတယ်။

S-72

```
import pikepdf
pdf = pikepdf.Pdf.open('/home/dev/Pictures/script_2.pdf')
docinfo = pdf.docinfo
for key, value in docinfo.items():
    print(key, ":", value)
✓ 0.4s
/CreationDate : D:20230717032503-04'00'
/Creator : Writer
/Producer : LibreOffice 7.3
```

Description

Line1: pikepdf နဲ့ပတ်သတ်တာတွေအသုံးပြုဖို့အတွက် import လုပ်ပြောင်းပါတယ်။

Line2: pdf file နဲ့ connect လုပ်လိုက်တာဖြစ်ပါတယ်။

Line3: pdf doc ရဲ့ info တွေကို ယူလိုက်တာဖြစ်ပါတယ်။

Line 4 and 5 : ရှိတဲ့ metadata တွေကို output ထုတ်ပြလိုက်တာဖြစ်ပါတယ်။

S-73

```
import keyboard
while True:
    if keyboard.is_pressed("a"):
        print("You pressed 'a'.")
        break
ⓧ 0.2s
```

Description

Keyboard module ကို အသုံးပြုထားပါတယ်။ program က အရ run နေတဲ့အချိန်အတွင်း a ကို နှိပ်လိုက်ရင် a လို့ output ထုတ်ပြီးရပ်လိုက်မှာဖြစ်ပါတယ်။ ဒီနည်းနဲ့ user နှိပ်လိုက်တဲ့ key တွေကို track လုပ်လို့ရပါတယ်။ နှိပ်သမျှ ကို သိချင်တယ်ဆိုရတော့ read_key method ကို အသုံးပြုလို့ရပါတယ်။

S-74

```
from zipfile import ZipFile
with ZipFile("/home/dev/Pictures/history.zip", 'r') as file_location:
    file_location.extractall(path="/home/dev/Pictures/history")
✓ 0.0s
```

Description

Python ရဲ့ Zipfile ကို အသုံးပြုထားပါတယ်။ Line2 ကတော့ zip file ရှိတဲ့ file location နဲ့ချိတ်ဆက်ပေးတာဖြစ်ပြီး Line 2 ကတော့ ထို file extract ဖြည့်လိုက်တဲ့ location ပါ။

S-75

```
import nmap
nm = nmap.PortScanner()
target = '64.233.170.100'
● nm.scan(target, '441-443')
✓ 1m 14.1s

{'nmap': {'command_line': 'nmap -oX - -p 441-443 -sV 64.233.170.100',
'scaninfo': {'tcp': {'method': 'connect', 'services': '441-443'}}},
'scanstats': {'timestr': 'Sun Sep 24 19:24:33 2023',
'elapsed': '74.05',
'uphosts': '1',
'downhosts': '0'}}
```

Description

Nmap module ကိုအသုံးပြုပြီးတည်ဆောက်ထားတဲ့ simple scanner တစ်ခုပါ။ တစ်ခုသိထားရမှာက data တွေကို json format အဖြစ် output ထုတ်ထားတာဖြစ်ပါတယ်။

S-76